

REMARKS

Claims 1-31 are pending in the present application. Claims 1-8, 10-18, and 20-31 have been amended to delete unnecessary language, to explicitly recite what was implicit, to further clarify the invention, and/or to correct informalities. Amendments to the claims are not intended to limit the scope of the invention. Support for the claim amendments can be found, for instance, on pages 18-20 of the specification. Applicant reserves the right to pursue any of the amended claims in their original form in a continuation application. No new matter has been added.

Objections to the Claims

Claims 1, 11, 21, 24, and 27 have been objected to on the basis of informalities. In particular, the Office action objected to the use of the term “for,” which the Office action asserts only indicates “‘system ability’ and/or ‘intended use’ and do(es) not hold patentable weight” (March 29, 2006 Office action, pg. 2).

Claims 1, 11, 21, and 24 have been amended to recite “storing an access control statement in a cache entry . . .” rather than “providing a cache for . . .” Claim 27 has been amended to recite “a database management system operable to receive a query . . .” and “a cache . . . operable to store an access control statement . . .” rather than “a database management system for . . .” and “a cache . . . for . . .”

Accordingly, based at least on the reasons above, Applicant respectfully requests withdrawal of the objections to claims 1, 11, 21, 24, and 27.

Objections to the Specification

The specification has been objected to. In particular, the Office action states:

The specification is objected to as failing to provide proper antecedent basis for the claimed subject matter. See 37 CFR 1.75(d)(1) and MPEP § 608.01(o). Correction of the following is required: claims 11 and 24 recite the limitation “computer readable medium”. There is insufficient antecedent basis for this claim. The specification does not teach any definition of “computer readable medium” or “media” in general. Appropriate correction is required.

(March 29, 2006 Office action, pg. 3).

Applicant respectfully disagrees with the Office action’s assertion that the term “computer readable medium” needs to be taught in the specification. One of skill in the art will readily understand the meaning and scope of the term “computer readable medium.” Further, “computer readable medium” claims are recognized as an accepted type of claim, similar to “method,” “system,” and “device” claims. Specifically, MPEP § 2106 states:

[A] claimed computer-readable medium encoded with a computer program is a computer element which defines structural and functional interrelationships between the computer program and the rest of the computer which permit the computer program’s functionality to be realized, and is thus statutory.

(M.P.E.P. § 2106.IV.B.1(a), 8th ed., 4th rev.)

Therefore, based at least on the reasons above, Applicant respectfully requests withdrawal of the objections to claims 11 and 24.

§ 101 Rejections

Claims 1, 11, 21, 24, and 27 have been rejected under 35 U.S.C. § 101 as being directed to non-statutory subject matter. The Office action states:

Claims 1, 11, 21, 24, and 27 are rejected under 35 U.S.C. § 101 because the result of “determining” does not produce a tangible result. The step of determining as recited in the claim is nothing more than a thought or a computation within a processor. What is determined is neither used nor made available for use to enable its usefulness in the disclosed practical application to be realized.

(March 29, 2006 Office action, pg. 3).

Claims 1 and 11 have been amended to recite “granting or denying access to the node based on the access control statement in the cache entry for the path associated with the node.” Claims 21 and 24 have been amended to recite “evaluating a value expression for the path associated with the node to produce a result.” Claim 27 has been amended to recite “the database management system is . . . operable . . . to grant or deny access to the node based on the access control statement in the cache entry for the path associated with the node.”

Applicant respectfully submits that a tangible result is produced in each of claims 1, 11, 21, 24, and 27, as amended, since permitting an action to take place or preventing an action from taking place are useful, concrete, and tangible.

Hence, based at least on the above reasons, Applicant respectfully submits that claims 1, 11, 21, 24, and 27 are directed to statutory subject matter.

§ 102 Rejections

Claims 1-31 have been rejected under 35 U.S.C. § 102(b) as being anticipated by “Design and Implementation of an Access Control Processor for XML Documents” by Ernesto Damiani, et. al. (hereinafter “Damiani”).

Claim 1, as amended, recites “storing an access control statement in a cache entry for a path associated with a node of the plurality of nodes.” The Office action states:

Damiani et al. teaches . . . providing a cache (see page 68, section 5.3, “Performance and caching”) for temporarily storing a cache entry for a path associated with a node of the plurality of nodes (see page 65, section 3.1, “Identifying authorization objects via path expressions”)

(March 29, 2006 Office action, pgs. 4-5).

The first cited passage of Damiani states:

In a complex server environment, performance and memory usage are critical issues. Moreover, the processing requirement for XML parsing, transformation, document processing and formatting are particularly heavy (Fig. 5). For this reason, a special cache system is needed, in order to cache dynamically created pages. Caches of this kind are already available for XSLT processors which store their stylesheets in a pre-parsed form [12]. A cache for labeled documents is an important part of our system. When the request comes, the cache is searched. If an instance of the requested document for the same subject is found in the cache, then the cache copy is served. Otherwise, the document is parsed, labeled, transformed, unparsed and sent to the client; also, the transformed document is stored in the cache. Whenever authorizations are changed the whole cache is emptied. This technique allows dynamically generated pages (for example, XML documents created by querying a database) to be transformed and cached. Assuming that the frequency of requests is higher than that of resource changes, the cache may greatly reduce the total server load. The efficiency gain is particularly relevant when authorizations are specified with respect to a limited number of groups, as it may be the case for Internet-based servers. Moreover, the cache system can be based on a persistent object storage system which is able to save stored objects in a persistent state that outlives the module execution. This technique can be effectively used for pages that are very expensive to generate and last very long without changes, such as compiled server pages.

(Sec. 5.3, pgs. 68-69 of Damiani). Thus, the first cited passage only discusses storing XML documents in a cache in order to reduce server load.

In contrast, claim 1 recites “storing an access control statement in a cache entry for a path associated with a node of the plurality of nodes.” Thus, in claim 1, it is the “access control statement” that is being stored in the “cache entry,” not the “structured document.”

The second cited passage of Damiani states:

In the traditional Web security setting, Uniform Resource Identifiers (URI) [2] are used to denote the resources to be protected. Each document and DTD is characterized by a single URI. As we go to a finer level of granularity we need to reference specific elements and attributes in documents. Elements/attributes in a document can be referenced by means of *path expressions*. A straightforward way of writing path expressions is by using the XPath language [20]. The reason for this choice is that several tools are currently available which can be easily reused to produce a functioning system. XPath expressions make reference to the tree organization of documents/DTDs which is obtained in a simple way by interpreting elements and attributes as children of the element in which they are directly contained. Each element and attribute can be then referenced by means of the tree path that must be followed to reach it. An XPath on an XML document tree is a sequence of element names or predefined functions separated by the character / (slash): $l_1 / l_2 / \dots / l_n$. For instance, path expression /division/about_div/member denotes the nodes of the member element which are children of about_div elements, which are children of division elements. Path expressions can be *absolute* or *relative*. Absolute path expressions, prefixed by a slash character, start from the root of the document. Relative path expressions, which start with an element name, describe a path whose initial point is any element in the document.

A very interesting characteristic of path expressions which very conveniently increases the expressiveness of authorizations is the support of conditions. Conditions associated with a path expression refine the set of nodes matching the path expression. Conditions may impose constraints on element contents (i.e., the ‘text’ of elements) or on names and values of attributes. A condition can follow any label in a path expression and is identified as such by enclosing it between square brackets. Given a path expression $l_1 / l_2 / \dots / l_n$, a condition on label l_i restricts the application of the path expression only to those node(s) l_i for which the condition evaluates to true.

(Sec. 3.1, pg. 65 of Damiani). Although the second cited passage mentions paths, it also fails to disclose, teach, or suggest “storing an access control statement in a cache entry for a path associated with a node of the plurality of nodes,” as recited in claim 1. In fact, the term “cache” is not even mentioned in the second cited passage.

Therefore, Damiani does not disclose, teach, or suggest “storing an access control statement in a cache entry for a path associated with a node of the plurality of nodes,” as recited in claim 1.

Claim 1, as amended, also recites “granting or denying access to the node based on the access control statement in the cache entry for the path associated with the node.” The Office action states:

Damiani et al. teaches . . . determining whether to grant access to the node based on the cache entry (see page 66, section 4, “Authorization enforcement”, lines 1-5).

(March 29, 2006 Office action, pgs. 4-5).

The cited passage of Damiani states:

For each possible requester (user connected from a certain location) and document, the authorizations on the document applicable to the requester describe what information can or cannot be returned to the requester. Hence, given the request from a subject to access a document, the joint application of the DTD-level and document-level authorizations applicable to the subject will produce a custom *view* on the document, including only the information that a particular requester is entitled to see. The access control process must therefore evaluate the authorizations applicable to an access request to compute such a view.

(Sec. 4, pg. 66, lines 1-13 of Damiani). Hence, the cited passage does not disclose, teach, or suggest “granting or denying access to the node based on the access control statement in the

cache entry for the path associated with the node,” as recited in claim 1, since it only discusses using DTD-level and document-level authorizations to determine what a requester sees.

Accordingly, based at least on the reasons above, Applicant respectfully submits that claim 1, and the claims that depend therefrom, are not anticipated by Damiani. Given that claims 11, 21, 24, and 27 each recites elements similar to those of claim 1, it is respectfully submitted that those claims, and the claims that depend therefrom, are not anticipated by Damiani for at least the same reasons.

CONCLUSION

On the basis of the above remarks, reconsideration and allowance of the claims is believed to be warranted and such action is respectfully requested. If the Examiner has any questions or comments, the Examiner is respectfully requested to contact the undersigned at the number listed below.

Respectfully submitted,
SAWYER LAW GROUP LLP

Dated: June 27, 2006


Erin C. Ming
Attorney for Applicant
Reg. No. 47,797
(650) 475-1449